Optimus-1 : Hybrid Multimodal Memory Empowered Agents Excel in Long-Horizon Tasks

Zaijing Li[1 2], Yuquan Xie[1], Rui Shao[1*], Gongwei Chen[1], Dongmei Jiang[2], Liqiang Nie[1*]
[1]Harbin Institute of Technology, Shenzhen [2]Peng Cheng Laboratory
{lzj14011,xieyuquan20016}@gmail.com, {shaorui,nieliqiang}@hit.edu.cn
https://cybertronagent.github.io/Optimus-1.github.io/

Abstract

Building a general-purpose agent is a long-standing vision in the field of artificial intelligence. Existing agents have made remarkable progress in many domains, yet they still struggle to complete long-horizon tasks in an open world. We attribute this to the lack of necessary world knowledge and multimodal experience that can guide agents through a variety of long-horizon tasks. In this paper, we propose a ***Hybrid Multimodal Memory*** *module to address the above challenges. It* ***1)*** *transforms knowledge into* ***Hierarchical Directed Knowledge Graph*** *that allows agents to explicitly represent and learn world knowledge, and* ***2)*** *summarises historical information into* ***Abstracted Multimodal Experience Pool*** *that provide agents with rich references for in-context learning. On top of the Hybrid Multimodal Memory module, a multimodal agent, Optimus-1, is constructed with dedicated* ***Knowledge-guided Planner*** *and* ***Experience-Driven Reflector****, contributing to a better planning and reflection in the face of long-horizon tasks in Minecraft. Extensive experimental results show that Optimus-1 significantly outperforms all existing agents on challenging long-horizon task benchmarks, and exhibits near human-level performance on many tasks. In addition, we introduce various Multimodal Large Language Models (MLLMs) as the backbone of Optimus-1. Experimental results show that Optimus-1 exhibits strong generalization with the help of the Hybrid Multimodal Memory module, outperforming the GPT-4V baseline.*

1. Introduction

Optimus Prime faces complex tasks alongside humans in Transformers to protect the peace of the planet. Creating an agent [13, 43] like Optimus that can perceive, plan, reflect, and complete long-horizon tasks in an open world has been a longstanding aspiration in the field of artificial intelligence [22, 27, 35, 36, 57]. Early research developed simple policy through reinforcement learning [7] or imitation learning [1, 25]. A lot of work [46, 49] have utilized Large Language Models (LLMs) as action planners for agents, generating executable sub-goal sequences for low-level action controllers. Further, recent studies [33, 51] employed Multimodal Large Language Models (MLLMs) [4, 38, 55] as planner and reflector. Leveraging the powerful instruction-following and logical reasoning capabilities of (Multimodal) LLMs [24], LLM-based agents have achieved remarkable success across multiple domains [9, 10, 14, 54]. Nevertheless, the ability of these agents to complete long-horizon tasks still falls significantly short of human-level performance.

According to relevant studies [28, 41, 45], the human ability to complete long-horizon tasks in an open world relies on long-term memory storage, which is divided into knowledge and experience. The storage and utilization of knowledge and experience play a crucial role in guiding human behavior and enabling humans to adapt flexibly to their environments in order to accomplish long-horizon tasks. Inspired by this theory, we summarize the challenges faced by current agents as follows:

Insufficient Exploration of Structured Knowledge: Structured knowledge, encompassing open world rules, object relationships, and interaction methods with the environment, is essential for agents to complete complex tasks [34, 43]. However, MLLMs such as GPT-4V [1] lack sufficient knowledge in Minecraft. Existing agents [1, 7, 25] only learn dispersed knowledge from video data and are unable to efficiently represent and learn this structured knowledge, rendering them incapable of performing complex tasks.

Lack of Multimodal Experience: Humans derive successful strategies and lessons from information on historical experience [8, 32], which assists them in tackling current complex tasks. In a similar manner, agents can benefit from in-context learning with experience demonstrations [42, 53].

*Corresponding authors

[1]https://openai.com/index/gpt-4v-system-card/

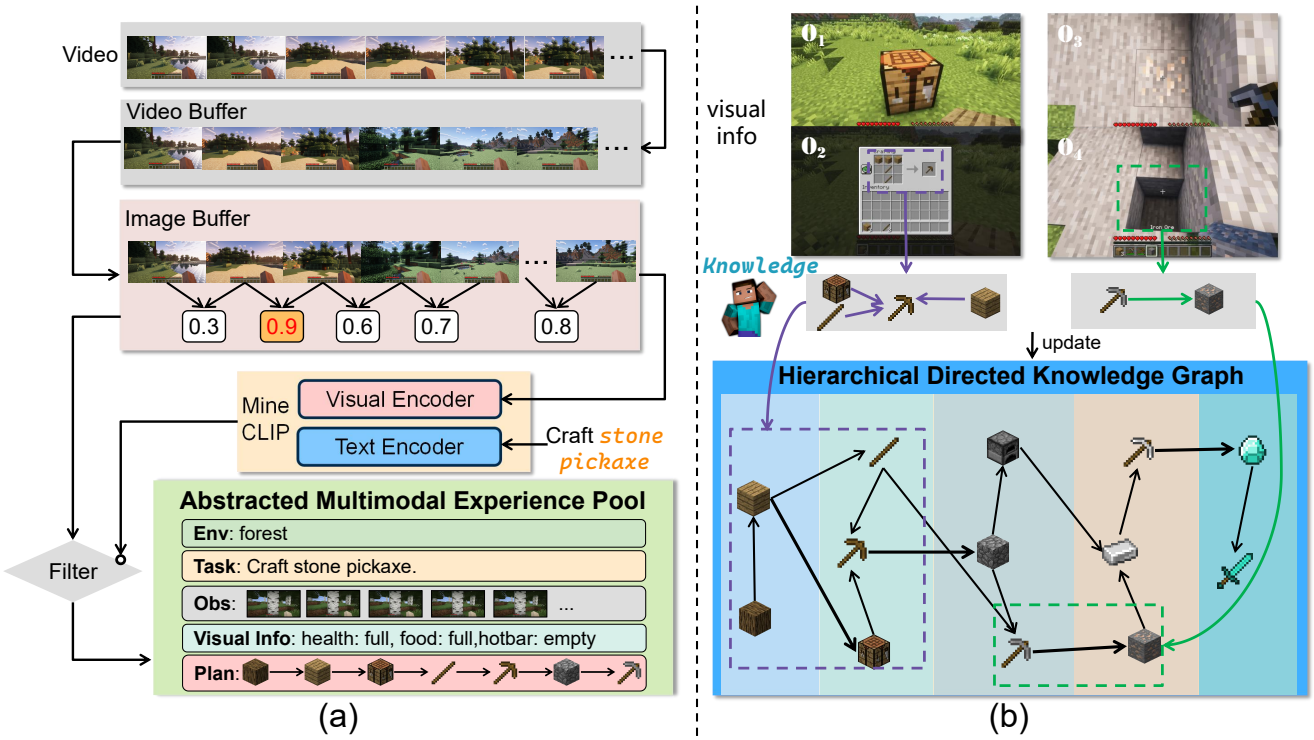


Figure 1. **(a)** Extraction process of multimodal experience. The frames are filtered through video buffer and image buffer, then MineCLIP [7] is employed to compute the visual and sub-goal similarities and finally they are stored in Abstracted Multimodal Experience Pool. **(b)** Overview of Hierarchical Directed Knowledge Graph. Knowledge is stored as a directed graph, where its nodes represent objects, and directed edges point to materials that can be crafted by this object.

However, existing agents [33, 46, 50] only consider unimodal information, which prevents them from learning from multimodal experience as humans do.

To address the aforementioned challenges, we propose **Hybrid Multimodal Memory** module that consists of **Hierarchical Directed Knowledge Graph** (HDKG) and **Abstracted Multimodal Experience Pool** (AMEP). For HDKG, we map the logical relationships between objects into a directed graph structure, thereby transforming knowledge into high-level semantic representations. HDKG efficiently provides the agent with the necessary knowledge for task execution, without requiring any parameter updates. For AMEP, we dynamically summarize and store the multimodal information (*e.g.,* environment, agent state, task plan, video frames, etc.) from the agent's task execution process, ensuring that historical information contains both a global overview and local details. Different from the method of directly storing successful cases as experience [51], AMEP considers both successful and failed cases as references. This innovative approach of incorporating failure cases into in-context learning significantly enhances the performance of the agent.

On top of the Hybrid Multimodal Memory module, we construct a multimodal composable agent, **Optimus-1**, which consists of Knowledge-Guided Planner, Experience-Driven Reflector, and Action Controller. To enhance the ability of agents to cope with complex environments and long-horizon tasks, Knowledge-Guided Planner incorporates visual observation into the planning phase, leveraging HDKG to capture the knowledge needed. This allows the agent to efficiently transform tasks into executable sub-goals. Action Controller takes the sub-goal and the current observation as inputs and generates low-level actions, interacting with the game environment to update the agent's state. In open-world complex environments, agents are prone to be erroneous when performing long-horizon tasks. To address this, we propose Experience-Driven Reflector, which is periodically activated to retrieve relevant multimodal experiences from AMEP. This encourages the agent to reflect on its current actions and refine the plan.

We validate the performance of Optimus-1 in Minecraft, a popular open-world game environment. Experimental results show that Optimus-1 exhibits remarkable performance on long-horizon tasks, representing up to 30% improvement over existing agents. Moreover, we introduce various Multimodal Large Language Models (MLLMs) as the backbone of Optimus-1. Experimental results show that Optimus-1 has a 2 to 6 times performance improvement with the help of Hy-

brid Multimodal Memory, outperforming powerful GPT-4V baseline on lots of long-horizon tasks. Additionally, we verified that the plug-and-play Hybrid Multimodal Memory can drive Optimus-1 to incrementally improve its performance in a self-evolution manner. The extensive experimental results show that Optimus-1 makes a major step toward a general agent with a human-like level of performance. Main contributions of our paper:

- We propose **Hybrid Multimodal Memory** module which is composed of HDKG and AMEP. HDKG helps the agent make the planning of long-horizon tasks efficiently. AMEP provides refined historical experience and guides the agent to reason about the current situation state effectively.
- On top of the Hybrid Multimodal Memory module, we construct **Optimus-1**, which consists of **Knowledge-Guided Planner**, **Experience-Driven Reflector**, and Action Controller. Optimus-1 outperforms all baseline agents on long-horizon task benchmarks, and exhibits capabilities close to the level of human players.
- Driven by Hybrid Multimodal Memory, various MLLM-based Optimus-1 have demonstrated 2 to 6 times performance improvement, demonstrating the generalization of Hybrid Multimodal Memory.

2. Optimus-1

In this section, we first elaborate on how to implement the Hybrid Multimodal Memory in Sec 2.1. As a core innovation, it plays a crucial role in enabling Optimus-1 to execute long-horizon tasks. Next, we give an overview of Optimus-1 framework (Sec 2.2), which consists of Hybrid Multimodal Memory, Knowledge-Guided Planner, Experience-Driven Reflector, and Action Controller. Finally, we introduce a non-parametric learning approach to expand the hybrid multimodal memory (Sec 2.3), thereby enhancing the success rate of task execution for Optimus-1.

2.1. Hybrid Multimodal Memory

In order to endow agent with a long-term memory storage mechanism [28, 45], we propose the Hybrid Multimodal Memory module, which consists of Abstracted Multimodal Experience Pool (AMEP) and Hierarchical Directed Knowledge Graph (HDKG).

2.1.1. Abstracted Multimodal Experience Pool

Relevant studies [15, 17, 23, 29] highlight the importance of historical information for agents completing long-horizon tasks. Minedojo [7] and Voyager [46] employed unimodal storage of historical information. Jarvis-1 [51] used a multimodal experience mechanism that stores task planning and visual information without summarization, posing challenges to storage capacity and retrieval speed. To address this issue, we propose AMEP, which aims to dynamically summarize all multimodal information during task execution. It preserves the integrity of long-horizon data while enhancing storage and retrieval efficiency.

Specifically, as depicted in Figure 1, to conduct the static visual information abstraction, the video stream captured by Optimus-1 during task execution is first input to a video buffer, filtering the stream at a fixed frequency of 1 frame per second. Based on the filtered video frames, to further perform a dynamic visual information abstraction, these frames are then fed into an image buffer with a window size of 16, where the image similarity is dynamically computed and final abstracted frames are adaptively updated. To align such abstracted visual information with the corresponding textual sub-goal, we then utilize MineCLIP [7], a pre-trained video-text alignment model, to calculate their multimodal correlation. When this correlation exceeds a threshold, the corresponding image buffer and textual sub-goal are saved as multimodal experience into a pool. Finally, we further incorporate environment information, agent initial state, and plan generated by Knowledge-Guided Planner, into such a pool, which forms the AMEP. In this way, we consider the multimodal information of each sub-goal, and summarise it to finally compose the multimodal experience of the given task.

2.1.2. Hierarchical Directed Knowledge Graph

In Minecraft, mining and crafting represent a complex knowledge network crucial for effective task planning. For instance, crafting a diamond sword requires two diamonds and one wooden stick, while mining diamonds requires an iron pickaxe, which involving further materials and steps. Such knowledge is essential for an agent's ability to perform long-horizon complex tasks. Instead of implicit learning through fine-tuning [33, 59], we propose HDKG, which transforms knowledge into a graph representation. It enables the agent to perform explicit learning by retrieving information from the knowledge graph.

As shown in the Figure 1, we transform knowledge into a graph $\mathcal{D}(\mathcal{V}, \mathcal{E})$, where nodes set $\mathcal{V}$ represent objects, and directed edges set $\mathcal{E}$ point to nodes that can be crafted by this object. An edge $e \in \mathcal{E}$ in the $\mathcal{D}$ can be represented as $e = (u, v)$, where $u, v \in \mathcal{V}$. The directed graph efficiently stores and updates knowledge. For a given object x, retrieving the corresponding node allows extraction of a sub-graph $\mathcal{D}_j(\mathcal{V}_j, \mathcal{E}_j) \in \mathcal{D}$, where nodes set $\mathcal{V}_j$ and edges set $\mathcal{E}_j$ can be formulated as:

$$\mathcal{V}_j = \{v \in \mathcal{V} \mid x\} \tag{1}$$

$$\mathcal{E}_j = \{e = (u, v) \in \mathcal{V} \mid u \in \mathcal{V}_j \cup v \in \mathcal{V}_j\}, \tag{2}$$

Then by topological sorting, we can get all the materials and their relationships needed to complete the task. This knowledge is provided to the Knowledge-Guided Planner as a way to generate a more reasonable sequence of sub-

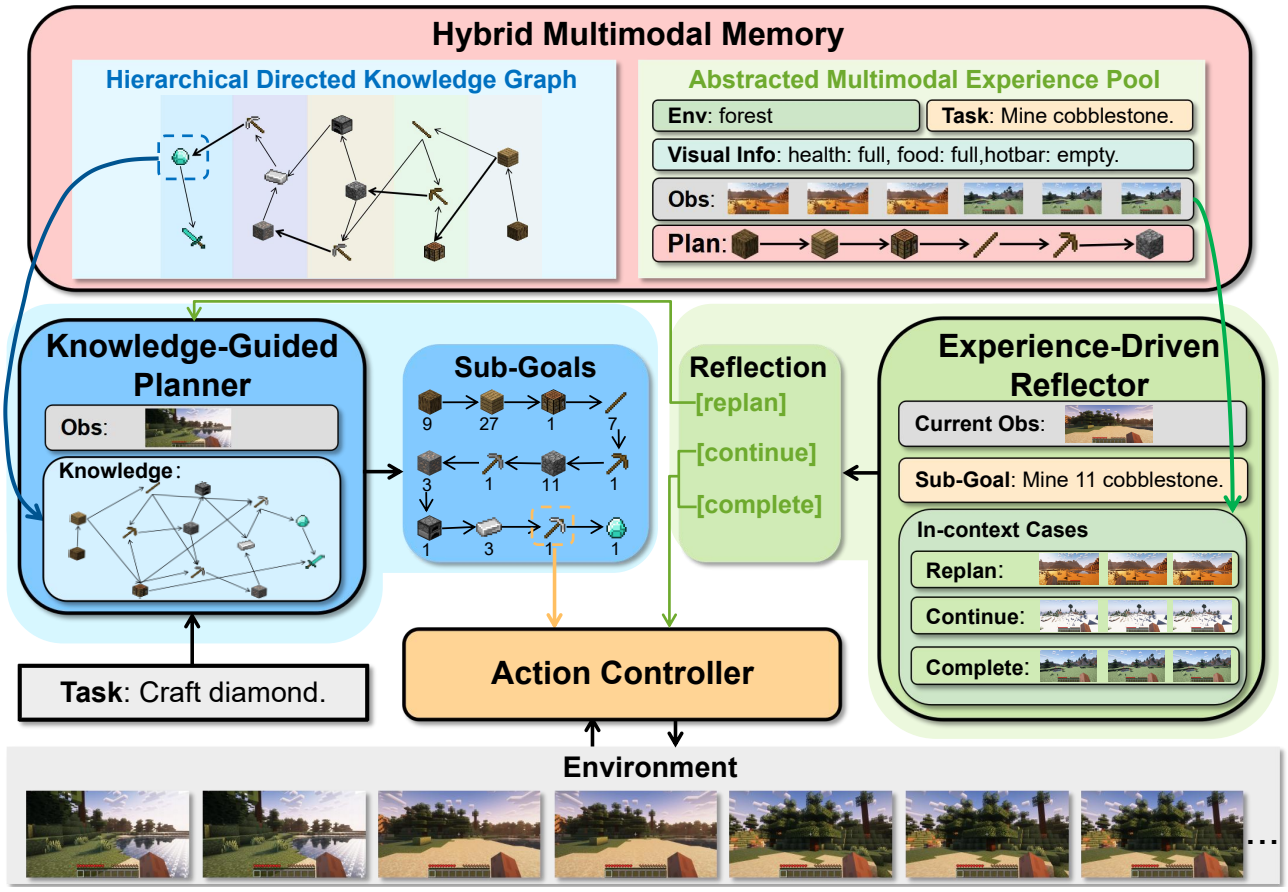


Figure 2. Overview framework of our Optimus-1. Optimus-1 consists of Knowledge-Guided Planner, Experience-Driven Reflector, Action Controller, and Hybrid Multimodal Memory architecture. Given the task "craft stone sword", Optimus-1 incorporates the knowledge from HDKG into Knowledge-Guided Planning, then Action Controller generates low-level actions. Experience-Driven Reflector is periodically activated to introduce multimodal experience from AMEP to determine if the current task can be executed successfully. If not, it will ask the Knowledge-Guided Planner to refine the plan.

goals. With HDKG, we can significantly enhance the world knowledge of the agent in a train-free manner.

2.2. Optimus-1: Framework

Relevant studies indicate that the human brain is essential for planning and reflection, while the cerebellum controls low-level actions, both crucial for complex tasks [39, 40]. Inspired by this, we divide the structure of Optimus-1 into Knowledge-Guided Planner, Experience-Driven Reflector, and Action Controller. In a given game environment with a long-horizon task, the Knowledge-Guided Planner senses the environment, retrieves knowledge from HDKG, and decomposes the task into executable sub-goals. The action controller then sequentially executes these sub-goals. During execution, the Experience-Driven Reflector is activated periodically, leveraging historical experience from AMEP to assess whether Optimus-1 can complete the current sub-goal. If not, it instructs the Knowledge-Guided Planner to revise its plan. Through iterative interaction with the environment, Optimus-1 ultimately completes the task.

Knowledge-Guided Planner. Open-world environments vary greatly, affecting task execution. Previous approaches [50] using LLMs for task planning failed to consider the environment, leading to the failure of tasks. For example, an agent in a cave aims to catch fish. It lacks visual information to plan conditions on the current situation, such as "leave the cave and find a river". Therefore, we integrate environmental information into the planning stage. Unlike Jarvis-1 [51] and MP5 [33], which convert observation to textual descriptions, Optimus-1 directly employs observation as visual conditions to generate environment-related plans, *i.e.,* sub-goal sequences. This results in more comprehensive and reasonable planning. More importantly, Knowledge-Guided Planner retrieves the knowledge needed to complete the task from HDKG, allowing task planning to be done once, rather than generating the next step in each iteration. Given the

task t, observation o, the sub-goals sequence g_1, g_2, g_3, ..., g_n can be formulated as:

$$g_1, g_2, g_3, ..., g_n = p_\theta(o, t, p_\eta(t)), \tag{3}$$

where n is the number of sub-goals, p_η denotes sub-graph retrieved from HDKG, p_θ denotes MLLM. In this paper, we employ OpenAI's GPT-4V as Knowledge-Guided Planner and Experience-Driven Reflector. We also evaluate other alternatives of GPT-4V, such as open-source models like Deepseek-VL [26] and InternLM-XComposer2-VL [6] in Section 3.4.

Action Controller. It takes the sub-goal and the current observation as inputs and then generates low-level actions, which are control signals for the mouse and keyboard. Thus, it can interact with the game environment to update the agent's state and the observation. The formulation is as follows:

$$a_k = p_\pi(o, g_i), \tag{4}$$

where a_k denotes low-level action at time k, p_π denotes action controller. Unlike generating code [33, 46, 49], generating control actions for the mouse and keyboard [1, 3, 25, 51] more closely resembles human behavior. In this paper, we employ STEVE-1 [25] as our Action Controller.

Experience-Driven Reflector. The sub-goals generated by Knowledge-Guided Planner are interdependent. The failure of any sub-goal halts the execution of subsequent ones, leading to overall task failure. Therefore, a reflection module is essential to identify and rectify errors promptly. During task execution, the Experience-Driven Reflector activates at regular intervals, retrieving historical experience from AMEP, and then analyzing the current state of Optimus-1. The reflection results of Optimus-1 are categorized as `COMPLETE`, `CONTINUE`, or `REPLAN`. `COMPLETE` indicates successful execution, prompting the action controller to proceed to the next sub-goal. `CONTINUE` signifies ongoing execution without additional feedback. `REPLAN` denotes failure, requiring the Knowledge-Guided Planner to revise the plan. The reflection r generated by Experience-Driven Reflector can be formulated as:

$$r = p_\theta(o, g_i, p_\epsilon(t)), \tag{5}$$

where p_ϵ denotes multimodal experience retrieved from AMEP. Experimental results in Section 3.3 demonstrate that the Experience-Driven Reflector significantly enhances the success rate of long-horizon tasks.

During task execution, even in cases where task failure necessitates `REPLAN`, multimodal experiences are stored in AMEP. Thus, during the reflection phase, Optimus-1 can retrieve the most relevant cases from each of the three scenarios `COMPLETE`, `CONTINUE`, and `REPLAN` from AMEP as references. Experimental Results in Section 3.3 demonstrate the effectiveness of this innovative method of incorporating failure cases into in-context learning.

2.3. Non-parametric Learning of Hybrid Multimodal Memory

To implement the Hybrid Multimodal Memory and enhance Optimus-1's capacity, we propose a non-parametric learning method named "free exploration-teacher guidance". In the free exploration phase, Optimus-1's equipment and tasks are randomly initialized, and it explores random environments, acquiring world knowledge through environmental feedback. For example, it learns that "a stone sword can be crafted with a wooden stick and two cobblestones", storing this in the HDKG. Additionally, successful and failed cases are stored in the AMEP, providing reference experience for the reflection phase. We initialize multiple Optimus-1, and they share the same HDKG and AMEP. Thus the memory is filled up efficiently. After free exploration, Optimus-1 has basic world knowledge and multimodal experience. In the teacher guidance phase, Optimus-1 needs to learn a small number of long-horizon tasks based on extra knowledge. For example, it learns "a diamond sword is obtained by a stick and two diamonds" from the teacher, then perform the task "craft diamond sword". During the teacher guidance phase, Optimus-1's memory is further expanded and it gains the experience of executing complete long-horizon tasks.

Unlike fine-tuning, this method enhances Optimus-1 incrementally without updating parameters, in a self-evolution manner. Starting with an empty Hybrid Multimodal Memory, Optimus-1 iterates between "free exploration-teacher guidance" learning and unseen task inference. With each iteration, its memory capacity grows, enabling mastery of tasks from easy to hard.

3. Experiments

3.1. Experiments Setting

Environment. To ensure realistic gameplay like human players, we employ MineRL [11] with Minecraft 1.16.5 as our simulation environment. The agent operates at a fixed speed of 20 frames per second and only interacts with the environment via low-level action control signals of the mouse and keyboard.

Benchmark. We constructed a benchmark of 67 tasks to evaluate the Optimus-1's ability to complete long-horizon tasks. As illustrated in **Appendix** Table 4, we divide the 67 Minecraft tasks into 7 groups according to recommended categories in Minecraft.

Baseline. We compare Optimus-1 with various agents, including GPT-3.5 [2], GPT-4V, DEPS [50], and Jarvis-1 [51] on the challenging long-horizon tasks benchmark. In addition, we employed 10 volunteers to perform the same task on the benchmark, and their average performance served as a human-level baseline. Note that we initialize Optimus-1 with

[2]https://openai.com/research/gpt-3.5

Table 1. Main Result of Optimus-1 on long-horizon tasks benchmark. We report the average success rate (SR), average number of steps (AS), and average time (AT) on each task group, the results of each task can be found in the **Appendix** D. Lower AS and AT metrics mean that the agent is more efficient at completing the task, while $+\infty$ indicates that the agent is unable to complete the task. Overall represents the average result on the five groups of Iron, Gold, Diamond, Redstone, and Armor.

Group	Metric	GPT-3.5	GPT-4V	DEPS	Jarvis-1	Optimus-1	Human-level
Wood	SR ↑	40.16	41.42	77.01	93.76	**98.60**	100.00
	AT ↓	56.39	55.15	85.53	67.76	**47.09**	31.08
	AS ↓	1127.78	1103.04	1710.61	1355.25	**841.94**	621.59
Stone	SR ↑	20.40	20.89	48.52	89.20	**92.35**	100.00
	AT ↓	135.71	132.77	138.71	141.50	**129.94**	80.85
	AS ↓	2714.21	2655.47	2574.30	2830.05	**2518.88**	1617.00
Iron	SR ↑	0.00	0.00	16.37	36.15	**46.69**	86.00
	AT ↓	$+\infty$	$+\infty$	944.61	722.78	**651.33**	434.38
	AS ↓	$+\infty$	$+\infty$	8892.24	8455.51	**6017.85**	5687.60
Gold	SR ↑	0.00	0.00	0.00	7.20	**8.51**	17.31
	AT ↓	$+\infty$	$+\infty$	$+\infty$	787.37	**726.35**	557.08
	AS ↓	$+\infty$	$+\infty$	$+\infty$	15747.13	**15527.07**	13141.60
Diamond	SR ↑	0.00	0.00	0.60	8.98	**11.61**	16.98
	AT ↓	$+\infty$	$+\infty$	1296.96	1255.06	**1150.98**	744.82
	AS ↓	$+\infty$	$+\infty$	23939.30	25101.25	**23019.64**	16237.54
Redstone	SR ↑	0.00	0.00	0.00	16.31	**25.02**	33.27
	AT ↓	$+\infty$	$+\infty$	$+\infty$	1070.42	**932.50**	617.89
	AS ↓	$+\infty$	$+\infty$	$+\infty$	17408.40	**12709.99**	12357.00
Armor	SR ↑	0.00	0.00	9.98	15.82	**19.47**	28.48
	AT ↓	$+\infty$	$+\infty$	997.59	924.60	**824.53**	551.30
	AS ↓	$+\infty$	$+\infty$	17951.95	16492.96	**16350.56**	11026.00
Overall	SR ↑	0.00	0.00	5.39	16.89	**22.26**	36.41

an empty inventory, while DEPS [50] and Jarvis-1 [51] have tools in their initial state. This makes it more challenging for Optimus-1 to perform the same tasks.

Evaluation Metrics. The agent always starts in survival mode, with an empty inventory. We conducted at least 30 times for each task using different world seeds and reported the average success rate (**SR**) to ensure fair and thorough evaluation. Additionally, we add the average steps (**AS**) and average time (**AT**) as evaluation metrics.

3.2. Experimental Results

The overall experimental results on benchmark are shown in Table 1, see **SR** for each task in **Appendix** D. Optimus-1 has a success rate near 100% on the Wood Group. Compared with Jarvis-1, Optimus-1 has 29.28% and 53.40% improvement on the Diamond Group and Redstone Group, respectively. Optimus-1 achieves the best performance and the shortest elapsed time among all task groups. It reveals the effectiveness and efficiency of our proposed Optimus-1 framework. Moreover, compared with all baselines, Optimus-1 performance was closer (average 5.37% improvement) to human levels on long-horizon task groups.

3.3. Ablation Study

We conduct extensive ablation experiments on 18 tasks, experiment setting can be found in **Appendix** Table 5. As

Table 2. Ablation study results. We report average success rate (SR) on each task group. `P.`, `R.`, `K.`, `E.` represent Planning, Reflection, Knowledge, and Experience, respectively.

Ablation Setting				Task Group				
P.	R.	K.	E.	Wood	Stone	Iron	Gold	Diamond
				14.29	0.00	0.00	0.00	0.00
✓				42.95	25.67	0.00	0.00	0.00
✓	✓			55.00	47.37	18.11	2.08	1.11
✓	✓		✓	73.53	64.20	24.19	3.08	1.86
✓	✓	✓		92.37	69.63	38.33	3.49	2.42
✓	✓	✓	✓	**97.49**	**94.26**	**53.33**	**11.54**	**9.59**

Table 3. Ablation study on AMEP. We report the average success rate (SR) on each task group. `Zero`, `Suc.`, and `Fail.` represent retrieving from AMEP without getting the case, getting the success case, and getting the failure case, respectively.

Ablation Setting			Task Group				
Zero	Suc.	Fai.	Wood	Stone	Iron	Gold	Diamond
✓			92.00	79.26	36.32	4.25	3.25
	✓		95.00	84.29	46.98	9.36	7.89
		✓	95.00	81.10	45.47	7.50	6.39
	✓	✓	**97.49**	**94.26**	**53.33**	**11.54**	**9.59**

shown in Table 2, we first remove Knowledge-Driven Planner and Experience-Driven Reflector, the performance of Optimus-1 on all task groups drops dramatically. It demonstrates the necessity of Knowledge-Guided Planner and Experience-Driven Reflector modules for performing long-horizon tasks. As for Hybrid Multimodal Memory, we remove HDKG from Optimus-1. Without the help of world knowledge, the performance of Optimus-1 decreased by an average of 20% across all task groups. We then removed AMEP, this resulted in the performance of Optimus-1 decreased by an average of 12%. Finally, we performed ablation experiments on the way of retrieving cases from AMEP. As shown in Table 3, without retrieving cases from AMEP, the success rate shows an average of 10% decrease across all groups. It reveals that this reflection mechanism, which considers both success and failure cases, has a significant impact on the performance of Optimus-1.

3.4. Generalization Ability

In this section, we explore an interesting issue: whether generic MLLMs can effectively perform various long-horizon complex tasks in Minecraft using Hybrid Multimodal Memory. As shown in Figure 3, We employ Deepseek-VL [26] and InternLM-XComposer2-VL [6] as Knowledge-Guided Planner and Experience-Driven Reflector. The experimental results show that the original MLLM has low performance on long-horizon tasks due to the lack of knowledge and experience of Minecraft. With the assistance of Hybrid Multimodal Memory, the performance of MLLMs has improved by 2 to 6 times across various task groups, outperforming the GPT-4V baseline. This encouraging result demonstrates the generalization of the proposed Hybrid Multimodal Memory.

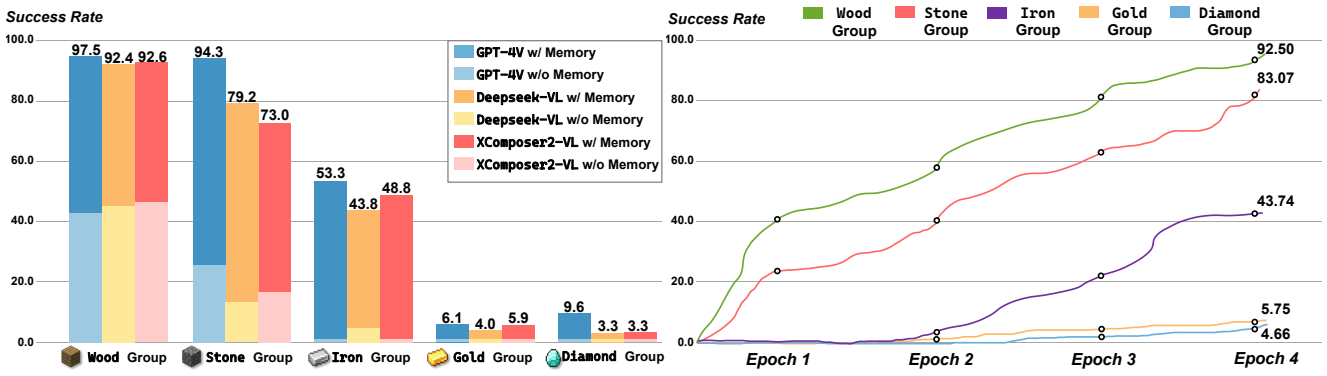
3.5. Self-Evolution via Hybrid Multimodal Memory

As shown in Section 2.3, we randomly initialize the Hybrid Multimodal Memory of Optimus-1, then update it multiple times by using the "free exploration-teacher guidance" learning method. We set the epoch to 4, and the number of learning tasks to 160. At each period, Optimus-1 performs free exploration on 150 tasks and teacher guidance learning on the remaining 10 tasks, we then evaluate Optimus-1's learning ability on the task groups same as ablation study. Experimental results are shown in Figure 3. It reveals that Optimus-1 keeps getting stronger through the continuous expansion of memory during the learning process of multiple periods. Moreover, it demonstrates that MLLM with Hybrid Multimodal Memory can incarnate an expert agent in a self-evolution manner [44].

4. Related Work

4.1. Agents in Minecraft

Earlier work [2, 3, 30, 56] introduced policy models for agents to perform simple tasks in Minecraft. MineCLIP [7] used text-video data to train a contrastive video-language model as a reward model for policy, while VPT [1] pre-trained on unlabelled videos but lacked instruction as input. Building on VPT and MineCLIP, STEVE-1 [25] added text input to generate low-level action sequences from human instructions and images. However, these agents struggle with complex tasks due to limitations in instruction comprehension and planning. Recent work [46, 49] incorporated LLMs as planning and reflection modules, but lacked visual information integration for adaptive planning. MP5 [33], MineDreamer [59], and Jarvis-1 [51] enhanced situation-aware planning by obtaining textual descriptions of visual information, yet lacked detailed visual data. Optimus-1 addresses these issues by directly using observation as situation-aware conditions in the planning phase, enabling more rational, visually informed planning. Additionally, unlike other agents requiring multiple queries for task refinement, Optimus-1 generates a complete and effective plan in one step with the help of HDKG. This makes Optimus-1 planning more efficient.



(a) Generalisation of Hybrid Multimodal Memory

(b) Self-Evolution

Figure 3. **(a)** With the help of Hybrid Multimodal Memory, various MLLM-based Optimus-1 have demonstrated 2 to 6 times performance improvement. **(b)** Illustration of the change in Optimus-1 success rate on the unseen task over 4 epochs.

4.2. Memory in Agents

In the agent-environment interaction process, memory is key to achieving experience accumulation [21], environment exploration [16], and knowledge abstraction [58]. There are two forms to represent memory content in LLM-based agents: textual form [15, 17, 31] and parametric form [5, 20, 29, 47]. In textual form, the information is explicitly retained and recalled by natural languages. In parametric form, the memory information [37] is encoded into parameters and implicitly influences the agent's actions. Recent work [12, 48, 52] has explored the long-term visual information storage [18, 19] and summarisation in MLLM. Our proposed hybrid multimodal memory module is plug-and-play and can provide world knowledge and multimodal experience for Optimus-1 efficiently.

5. Conclusion

In this paper, we propose Hybrid Multimodal Memory module, which consists of two parts: HDKG and AMEP. HDKG provides the necessary world knowledge for the planning phase of the agent, and AMEP provides the refined historical experience for the reflection phase of the agent. On top of the Hybrid Multimodal Memory, we construct the multimodal composable agent, Optimus-1, in Minecraft. Extensive experimental results show that Optimus-1 outperforms all existing agents on long-horizon tasks. Moreover, we validate that general-purpose MLLMs, based on Hybrid Multimodal Memory and without additional parameter updates, can exceed the powerful GPT-4V baseline. The extensive experimental results show that Optimus-1 makes a major step toward a general agent with a human-like level of performance.

References

[1] Bowen Baker, Ilge Akkaya, Peter Zhokov, Joost Huizinga, Jie Tang, Adrien Ecoffet, Brandon Houghton, Raul Sampedro, and Jeff Clune. Video pretraining (vpt): Learning to act by watching unlabeled online videos. *Advances in Neural Information Processing Systems*, 35:24639–24654, 2022. 1, 5, 7

[2] Shaofei Cai, Zihao Wang, Xiaojian Ma, Anji Liu, and Yitao Liang. Open-world multi-task control through goal-aware representation learning and adaptive horizon prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13734–13744, 2023. 7

[3] Shaofei Cai, Bowei Zhang, Zihao Wang, Xiaojian Ma, Anji Liu, and Yitao Liang. Groot: Learning to follow instructions by watching gameplay videos. In *The Twelfth International Conference on Learning Representations*, 2023. 5, 7

[4] Gongwei Chen, Leyang Shen, Rui Shao, Xiang Deng, and Liqiang Nie. Lion: Empowering multimodal large language model with dual-level visual knowledge. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. 1

[5] Nicola De Cao, Wilker Aziz, and Ivan Titov. Editing factual knowledge in language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6491–6506, 2021. 8

[6] Xiaoyi Dong, Pan Zhang, Yuhang Zang, Yuhang Cao, Bin Wang, Linke Ouyang, Xilin Wei, Songyang Zhang, Haodong Duan, Maosong Cao, et al. Internlm-xcomposer2: Mastering free-form text-image composition and comprehension in vision-language large model. *arXiv preprint arXiv:2401.16420*, 2024. 5, 7

[7] Linxi Fan, Guanzhi Wang, Yunfan Jiang, Ajay Mandlekar, Yuncong Yang, Haoyi Zhu, Andrew Tang, De-An Huang, Yuke Zhu, and Anima Anandkumar. Minedojo: Building open-ended embodied agents with internet-scale knowledge. *Advances in Neural Information Processing Systems*, 35: 18343–18362, 2022. 1, 2, 3, 7

[8] Mariel K Goddu and Alison Gopnik. The development of

human causal learning and reasoning. *Nature Reviews Psychology*, pages 1–21, 2024. 1

[9] Maitrey Gramopadhye and Daniel Szafir. Generating executable action plans with environmentally-aware language models. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3568–3575. IEEE, 2023. 1

[10] Izzeddin Gur, Hiroki Furuta, Austin Huang, Mustafa Safdari, Yutaka Matsuo, Douglas Eck, and Aleksandra Faust. A real-world webagent with planning, long context understanding, and program synthesis. *arXiv preprint arXiv:2307.12856*, 2023. 1

[11] William H Guss, Brandon Houghton, Nicholay Topin, Phillip Wang, Cayden Codel, Manuela Veloso, and Ruslan Salakhutdinov. Minerl: A large-scale dataset of minecraft demonstrations. *arXiv preprint arXiv:1907.13440*, 2019. 5

[12] Bo He, Hengduo Li, Young Kyun Jang, Menglin Jia, Xuefei Cao, Ashish Shah, Abhinav Shrivastava, and Ser-Nam Lim. Ma-lmm: Memory-augmented large multimodal model for long-term video understanding. *arXiv preprint arXiv:2404.05726*, 2024. 8

[13] Jiangyong Huang, Silong Yong, Xiaojian Ma, Xiongkun Linghu, Puhao Li, Yan Wang, Qing Li, Song-Chun Zhu, Baoxiong Jia, and Siyuan Huang. An embodied generalist agent in 3d world. *arXiv preprint arXiv:2311.12871*, 2023. 1

[14] Wenlong Huang, Pieter Abbeel, Deepak Pathak, and Igor Mordatch. Language models as zero-shot planners: Extracting actionable knowledge for embodied agents. In *International Conference on Machine Learning*, pages 9118–9147. PMLR, 2022. 1

[15] Ziheng Huang, Sebastian Gutierrez, Hemanth Kamana, and Stephen MacNeil. Memory sandbox: Transparent and interactive memory management for conversational agents. In *Adjunct Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, pages 1–3, 2023. 3, 8

[16] Yunfan Jiang, Agrim Gupta, Zichen Zhang, Guanzhi Wang, Yongqiang Dou, Yanjun Chen, Li Fei-Fei, Anima Anandkumar, Yuke Zhu, and Linxi Fan. Vima: General robot manipulation with multimodal prompts. In *NeurIPS 2022 Foundation Models for Decision Making Workshop*, 2022. 8

[17] Dacheng Li, Rulin Shao, Anze Xie, Ying Sheng, Lianmin Zheng, Joseph Gonzalez, Ion Stoica, Xuezhe Ma, and Hao Zhang. How long can context length of open-source llms truly promise? In *NeurIPS 2023 Workshop on Instruction Tuning and Instruction Following*, 2023. 3, 8

[18] Xiaojie Li, Shaowei He, Jianlong Wu, Yue Yu, Liqiang Nie, and Min Zhang. Mask again: Masked knowledge distillation for masked video modeling. In *Proceedings of the ACM International Conference on Multimedia*, page 2221–2232. ACM, 2023. 8

[19] Xiaojie Li, Jianlong Wu, Shaowei He, Shuo Kang, Yue Yu, Liqiang Nie, and Min Zhang. Fine-grained key-value memory enhanced predictor for video representation learning. In *Proceedings of the ACM International Conference on Multimedia*, page 2264–2274. ACM, 2023. 8

[20] Xiaojie Li, Yibo Yang, Xiangtai Li, Jianlong Wu, Yue Yu, Bernard Ghanem, and Min Zhang. Genview: Enhancing view quality with pretrained generative model for self-supervised learning. In *Proceedings of the European Conference on Computer Vision*. Springer, 2024. 8

[21] Xiaojie Li, Yibo Yang, Jianlong Wu, Bernard Ghanem, Liqiang Nie, and Min Zhang. Mamba-fscil: Dynamic adaptation with selective state space model for few-shot class-incremental learning. *arXiv preprint arXiv:2407.06136*, 2024. 8

[22] Zaijing Li, Fengxiao Tang, Ming Zhao, and Yusen Zhu. Emocaps: Emotion capsule based model for conversational emotion recognition. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1610–1618, 2022. 1

[23] Zaijing Li, Ting-En Lin, Yuchuan Wu, Meng Liu, Fengxiao Tang, Ming Zhao, and Yongbin Li. Unisa: Unified generative framework for sentiment analysis. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 6132–6142, 2023. 3

[24] Zaijing Li, Gongwei Chen, Rui Shao, Dongmei Jiang, and Liqiang Nie. Enhancing the emotional generation capability of large language models via emotional chain-of-thought. *arXiv preprint arXiv:2401.06836*, 2024. 1

[25] Shalev Lifshitz, Keiran Paster, Harris Chan, Jimmy Ba, and Sheila McIlraith. Steve-1: A generative model for text-to-behavior in minecraft. *Advances in Neural Information Processing Systems*, 2023. 1, 5, 7, 11

[26] Haoyu Lu, Wen Liu, Bo Zhang, Bingxuan Wang, Kai Dong, Bo Liu, Jingxiang Sun, Tongzheng Ren, Zhuoshu Li, Yaofeng Sun, et al. Deepseek-vl: towards real-world vision-language understanding. *arXiv preprint arXiv:2403.05525*, 2024. 5, 7

[27] Qi Lv, Xiang Deng, Gongwei Chen, Michael Y Wang, and Liqiang Nie. Decision mamba: A multi-grained state space model with self-evolution regularization for offline rl. In *NeurIPS*, 2024. 1

[28] Simon Makin. The amyloid hypothesis on trial. *Nature*, 559 (7715):S4–S4, 2018. 1, 3

[29] Eric Mitchell, Charles Lin, Antoine Bosselut, Chelsea Finn, and Christopher D Manning. Fast model editing at scale. In *International Conference on Learning Representations*, 2021. 3, 8

[30] Junhyuk Oh, Satinder Singh, Honglak Lee, and Pushmeet Kohli. Zero-shot task generalization with multi-task deep reinforcement learning. In *Proceedings of the 34th International Conference on Machine Learning*, pages 2661–2670. PMLR, 2017. 7

[31] Arka Pal, Deep Karkhanis, Manley Roberts, Samuel Dooley, Arvind Sundararajan, and Siddartha Naidu. Giraffe: Adventures in expanding context lengths in llms. *arXiv preprint arXiv:2308.10882*, 2023. 8

[32] Eileen Parkes. Scientific progress is built on failure. *Nature*, 10, 2019. 1

[33] Yiran Qin, Enshen Zhou, Qichang Liu, Zhenfei Yin, Lu Sheng, Ruimao Zhang, Yu Qiao, and Jing Shao. Mp5: A multi-modal open-ended embodied system in minecraft via active perception. *arXiv preprint arXiv:2312.07472*, 2023. 1, 2, 3, 4, 5, 7

[34] Maria Abi Raad, Arun Ahuja, Catarina Barros, Frederic Besse, Andrew Bolt, Adrian Bolton, Bethanie Brownfield,

Gavin Buttimore, Max Cant, Sarah Chakera, et al. Scaling instructable agents across many simulated worlds. *arXiv preprint arXiv:2404.10179*, 2024. 1

[35] Rui Shao, Xiangyuan Lan, Jiawei Li, and Pong C Yuen. Multi-adversarial discriminative deep domain generalization for face presentation attack detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10023–10031, 2019. 1

[36] Rui Shao, Tianxing Wu, and Ziwei Liu. Detecting and grounding multi-modal media manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6904–6913, 2023. 1

[37] Rui Shao, Tianxing Wu, Jianlong Wu, Liqiang Nie, and Ziwei Liu. Detecting and grounding multi-modal media manipulation and beyond. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. 8

[38] Leyang Shen, Gongwei Chen, Rui Shao, Weili Guan, and Liqiang Nie. Mome: Mixture of multimodal experts for generalist multimodal large language models. In *NeurIPS*, 2024. 1

[39] Shan H Siddiqi, Konrad P Kording, Josef Parvizi, and Michael D Fox. Causal mapping of human brain function. *Nature reviews neuroscience*, pages 361–375, 2022. 4

[40] JF Stein. Role of the cerebellum in the visual guidance of movement. *Nature*, pages 217–221, 1986. 4

[41] Tim Stuart, Andrew Butler, Paul Hoffman, Christoph Hafemeister, Efthymia Papalexi, William M Mauck, Yuhan Hao, Marlon Stoeckius, Peter Smibert, and Rahul Satija. Comprehensive integration of single-cell data. *cell*, 177(7):1888–1902, 2019. 1

[42] Quan Sun, Yufeng Cui, Xiaosong Zhang, Fan Zhang, Qiying Yu, Zhengxiong Luo, Yueze Wang, Yongming Rao, Jingjing Liu, Tiejun Huang, et al. Generative multimodal models are in-context learners. *arXiv preprint arXiv:2312.13286*, 2023. 1

[43] Weihao Tan, Ziluo Ding, Wentao Zhang, Boyu Li, Bohan Zhou, Junpeng Yue, Haochong Xia, Jiechuan Jiang, Longtao Zheng, Xinrun Xu, et al. Towards general computer control: A multimodal agent for red dead redemption ii as a case study. *arXiv preprint arXiv:2403.03186*, 2024. 1

[44] Zhengwei Tao, Ting-En Lin, Xiancai Chen, Hangyu Li, Yuchuan Wu, Yongbin Li, Zhi Jin, Fei Huang, Dacheng Tao, and Jingren Zhou. A survey on self-evolution of large language models. *arXiv preprint arXiv:2404.14387*, 2024. 7

[45] Deniz Vatansever, Jonathan Smallwood, and Elizabeth Jefferies. Varying demands for cognitive control reveals shared neural processes supporting semantic and episodic memory retrieval. *Nature communications*, 12(1):2134, 2021. 1, 3

[46] Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. Voyager: An open-ended embodied agent with large language models. *arXiv preprint arXiv:2305.16291*, 2023. 1, 2, 3, 5, 7

[47] Mengru Wang, Ningyu Zhang, Ziwen Xu, Zekun Xi, Shumin Deng, Yunzhi Yao, Qishen Zhang, Linyi Yang, Jindong Wang, and Huajun Chen. Detoxifying large language models via knowledge editing. *arXiv preprint arXiv:2403.14472*, 2024. 8

[48] Shihao Wang, Zhiding Yu, Xiaohui Jiang, Shiyi Lan, Min Shi, Nadine Chang, Jan Kautz, Ying Li, and Jose M Alvarez. Omnidrive: A holistic llm-agent framework for autonomous driving with 3d perception, reasoning and planning. *arXiv preprint arXiv:2405.01533*, 2024. 8

[49] Zihao Wang, Shaofei Cai, Guanzhou Chen, Anji Liu, Xiaojian Ma, and Yitao Liang. Describe, explain, plan and select: Interactive planning with large language models enables open-world multi-task agents. *arXiv preprint arXiv:2302.01560*, 2023. 1, 5, 7

[50] Zihao Wang, Shaofei Cai, Guanzhou Chen, Anji Liu, Xiaojian Ma, and Yitao Liang. Describe, explain, plan and select: Interactive planning with large language models enables open-world multi-task agents. *arXiv preprint arXiv:2302.01560*, 2023. 2, 4, 5, 6, 11

[51] Zihao Wang, Shaofei Cai, Anji Liu, Yonggang Jin, Jinbing Hou, Bowei Zhang, Haowei Lin, Zhaofeng He, Zilong Zheng, Yaodong Yang, et al. Jarvis-1: Open-world multi-task agents with memory-augmented multimodal language models. *arXiv preprint arXiv:2311.05997*, 2023. 1, 2, 3, 4, 5, 6, 7, 11

[52] Yuetian Weng, Mingfei Han, Haoyu He, Xiaojun Chang, and Bohan Zhuang. Longvlm: Efficient long video understanding via large language models. *arXiv preprint arXiv:2404.03384*, 2024. 8

[53] Xu Yang, Yongliang Wu, Mingzhuo Yang, Haokun Chen, and Xin Geng. Exploring diverse in-context configurations for image captioning. *Advances in Neural Information Processing Systems*, 36, 2024. 1

[54] Zhao Yang, Jiaxuan Liu, Yucheng Han, Xin Chen, Zebiao Huang, Bin Fu, and Gang Yu. Appagent: Multimodal agents as smartphone users. *arXiv preprint arXiv:2312.13771*, 2023. 1

[55] Qilang Ye, Zitong Yu, Rui Shao, Xinyu Xie, Philip Torr, and Xiaochun Cao. Cat: Enhancing multimodal large language model to answer questions in dynamic audio-visual scenarios. In *ECCV*, 2024. 1

[56] Haoqi Yuan, Chi Zhang, Hongcheng Wang, Feiyang Xie, Penglin Cai, Hao Dong, and Zongqing Lu. Skill reinforcement learning and planning for open-world long-horizon tasks. *arXiv preprint arXiv:2303.16563*, 2023. 7

[57] Haoyu Zhang, Meng Liu, Zixin Liu, Xuemeng Song, Yaowei Wang, and Liqiang Nie. Multi-factor adaptive vision selection for egocentric video question answering. In *Proceedings of the 41st International Conference on Machine Learning*, pages 59310–59328. PMLR, 2024. 1

[58] Zeyu Zhang, Xiaohe Bo, Chen Ma, Rui Li, Xu Chen, Quanyu Dai, Jieming Zhu, Zhenhua Dong, and Ji-Rong Wen. A survey on the memory mechanism of large language model based agents. *arXiv preprint arXiv:2404.13501*, 2024. 8

[59] Enshen Zhou, Yiran Qin, Zhenfei Yin, Yuzhou Huang, Ruimao Zhang, Lu Sheng, Yu Qiao, and Jing Shao. Minedreamer: Learning to follow instructions via chain-of-imagination for simulated-world control. *arXiv preprint arXiv:2403.12037*, 2024. 3, 7

A. Limitation and Future Work

In the framework of Optimus-1, we are dedicated to leverage proposed Hierarchical Directed Knowledge Graph and Abstracted Multimodal Experience Pool can be used to enhance the agent's ability to plan and reflect. For Action Controller, we directly introduce STEVE-1 [25] as a generator of low-level actions. However, limited by STEVE-1's ability to follow instructions and execute complex actions, Optimus-1 is weak in completing challenging tasks such as "beat ender dragon" and "build a house". Therefore, a potential future research direction is to enhance the instruction following and action generation capabilities of action controller.

In addition, most of the work, including Optimus-1, utilize a multimodal large language model for planning and reflection, which then drives an action controller to perform the task. Building an end-to-end vision-language-action agent will be future work.

B. Broader Impact

With the increasing capability level of Multimodal Large Language Models (MLLM) comes many potential benefits and also risks. On the positive side, we anticipate that the techniques that used to create Optimus-1 could be applied to the creation of helpful agents in robotics, video games, and the web. This plug-and-play architecture that we have created can be quickly adapted to different MLLMs, and the proposed methods also provide a viable solution for other application areas in the agent domain. However, on the negative side, it is imperative to acknowledge the inherent stochastic nature of MLLMs in text generation. If not addressed carefully, this could lead to devastating consequences for society. Prior to deploying MLLMs in conjunction with the Hybrid Multimodal Memory methodology, a comprehensive assessment of their potential risks must be undertaken. We hope that while the stakes are low, works such as ours can improve access to safety research on instruction-following models in multimodal agents domains.

C. Benchmark Suite

C.1. Benchmark

We constructed a benchmark of 67 tasks to evaluate Optimus-1's ability to complete long-horizon tasks in Minecraft. According to recommended categories in Minecraft, we have classified these tasks into 7 groups: Wood, Stone, Iron, Gold, Diamond, Redstone, Armor. The statistics for benchmark are shown in Table 4. Due to the varying complexity of these tasks, we adopt different maximum gameplay steps (Max. Steps) for each task. The maximum steps are determined by the average steps that human players need to complete the task. Due to the randomness of Minecraft, the world and initial spawn point of the agent could vary a lot. In our benchmark setting, We initialize the agent with an empty inventory, which makes it necessary for the agent to complete a series of sub-goals (mining materials, crafting tools) in order to perform any tasks. This makes every task challenging, even for human players.

Note that Diamonds are a very rare item that only spawns in levels 2 to 16 and have a 0.0846% chance of spawning in Minecraft 1.16.5. Diamonds are usually found near level 9, or in man-made or natural mines no higher than level 16. To mitigate the significant impact of diamond generation probability on the agent's likelihood of successfully completing the task, **we have adjusted the diamond generation probability to 20%, spawns in levels 2 to 16.** This setting applies to human players as well.

In the ablation study, we select the subset of our benchmark as the evaluation set (shown in Table 5). The environment setting is the same as the benchmark.

C.2. Baselines

Existing Baseline. On the one hand, we employ GPT-3.5 and GPT-4V as baseline, which are evaluated without integrating hybrid multimodal memory modules. During the planning phase, they generate a plan for the action controller based on task prompt (and observation). During the reflection phase, they generate reflection results in a zero-shot manner. On the other hand, we compare existing SOTA Agents [50, 51] in Minecraft.

Human-level Baseline. To better demonstrate agent's performance level in Minecraft, we hired 10 volunteers to play the game as a human-level baseline. The volunteers played the game with the same environment and settings, and every volunteer asked to perform the each task on the benchmark 10 times. Ultimately, we used the average scores of 10 volunteers as the human-level baseline. The results of the human-level baseline are shown in Table 1. To ensure the validity of the experiment, we ensured that each volunteer had at least 20 hours of Minecraft gameplay before conducting the experiment. For each volunteer, we pay $25 as reward.

D. Experimental Results

We list the results of each task on the benchmark below, with details including task name, sub-goal numbers, success rate (SR), average number of steps (AS), average time (AT), and eval times. All tasks are evaluated in Minecraft 1.16.5 Survival Mode. Note that each time Optimus-1 performs a task, we initial it with an empty initial inventory and a random start point. This makes it challenging for Optimus-1 to perform each task.

Experimental results show that Optimus-1's average task completion step (AS) is significantly lower than other baselines.

Table 4. Setting of 7 groups encompassing 67 Minecraft long-horizon tasks.

Group	Task Num.	Example Task	Max. Steps	Initial Inventory	Avg. Sub-goal Num.
Wooden	10	Craft a wooden axe	3600	Empty	5
Stone	9	Craft one stone pickaxe	7200	Empty	9
Iron	16	Craft a iron pickaxe	12000	Empty	13
Golden	6	Mine gold and smelt into golden ingot	36000	Empty	16
Redstone	6	Craft a piston	36000	Empty	17
Diamond	7	Dig down and mine a diamond	36000	Empty	15
Armor	13	Craft one iron helmet	36000	Empty	16

Table 5. We evaluate Optimus-1 on these tasks in ablation study which are the subset of our benchmark.

Group	Task	Sub-Goal Num.	Max. Step	Initial Inventory
Wooden	Craft a wooden axe	5	3600	Empty
	Craft a crafting table	3	3600	Empty
Stone	Craft a stone pickaxe	10	7200	Empty
	Craft a stone axe	10	7200	Empty
	Craft a furnace	9	7200	Empty
Iron	Craft a iron pickaxe	13	12000	Empty
	Craft a bucket	13	12000	Empty
	Craft a rail	13	12000	Empty
	Craft a iron sword	12	12000	Empty
	Craft a shears	12	12000	Empty
Golden	Craft a golden pickaxe	16	36000	Empty
	Craft a golden axe	16	36000	Empty
	Smelt a golden ingot	15	36000	Empty
Diamond	Craft a diamond pickaxe	15	36000	Empty
	Craft a diamond axe	16	36000	Empty
	Craft a diamond hoe	15	36000	Empty
	Craft a diamond sword	15	36000	Empty
	Dig down and mine a diamond	15	36000	Empty

Table 6. The results of Optimus-1 on various tasks in the Wood group. SR, AS, AT denote success rate, average number of steps, and average time (seconds), respectively.

Task	Sub-Goal Num.	SR	AS	AT(s)	Eval Times
Craft a wooden shovel	6	95.00	995.58	49.78	40
Craft a wooden pickaxe	5	100.00	1153.91	57.70	30
Craft a wooden axe	5	96.67	1010.28	50.51	30
Craft a wooden hoe	5	100.00	1042.80	52.14	30
Craft a stick	4	97.14	372.97	18.65	70
Craft a crafting table	3	98.55	448.63	22.43	69
Craft a wooden sword	5	100.00	1214.90	60.74	30
Craft a chest	4	100.00	573.80	28.69	30
Craft a bowl	4	100.00	744.30	37.21	30
Craft a ladder	4	100.00	820.30	41.02	30

Table 7. The results of Optimus-1 on various tasks in the Stone group. SR, AS, AT denote success rate, average number of steps, and average time (seconds), respectively.

Task	Sub-Goal Num.	SR	AS	AT(s)	Eval Times
Craft a stone shovel	8	90.32	2221.00	111.05	31
Craft a stone pickaxe	10	96.77	2310.09	115.50	31
Craft a stone axe	10	96.88	2112.59	105.63	32
Craft a stone hoe	8	94.64	2684.60	134.23	56
Craft a charcoal	9	88.57	3083.35	154.17	35
Craft a smoker	9	90.24	3118.89	155.94	41
Craft a stone sword	8	94.29	2067.92	103.40	35
Craft a furnace	9	93.75	2842.71	142.14	32
Craft a torch	8	85.71	2109.00	105.45	95

Table 8. The results of Optimus-1 on various tasks in the Iron group. SR, AS, AT denote success rate, average number of steps, and average time (seconds), respectively.

Task	Sub Goal Num.	SR	AS	AT(s)	Eval Times
Craft an iron shovel	13	54.79	5677.35	637.81	73
Craft an iron pickaxe	13	59.42	6157.39	591.81	69
Craft an iron axe	13	54.29	6026.26	676.97	70
Craft an iron hoe	13	52.70	6650.97	743.82	74
Craft a bucket	13	54.29	6124.61	591.35	70
Craft a hopper	14	46.67	7242.14	710.17	60
Craft a rail	13	42.19	6713.07	754.48	64
Craft an iron sword	12	57.14	5625.49	633.91	70
Craft a shears	12	53.62	5058.00	570.35	69
Craft a smithing table	12	44.93	5317.39	594.81	69
Craft a tripwire hook	13	48.57	4968.74	562.66	70
Craft a chain	13	44.93	5764.42	645.33	69
Craft an iron bars	12	42.00	6508.43	723.13	50
Craft an iron nugget	12	30.99	4697.23	525.29	71
Craft a blast furnace	14	25.71	7760.67	711.05	35
Craft a stonecutter	13	34.78	5993.38	675.52	46

Table 9. The results of Optimus-1 on various tasks in the Gold group. SR, AS, AT denote success rate, average number of steps, and average time (seconds), respectively.

Task	Sub Goal Num.	SR	AS	AT(s)	Eval Times
Craft a golden shovel	16	9.80	13734.75	686.74	51
Craft a golden pickaxe	16	13.75	9672.00	783.60	80
Craft a golden axe	16	4.44	10158.75	707.94	45
Craft a golden hoe	16	3.33	13120.50	756.03	27
Craft a golden sword	16	3.33	9792.00	789.60	26
Smelt and craft a golden ingot	15	16.42	9630.27	681.51	67

Table 10. The results of Optimus-1 on various tasks in the Diamond group. SR, AS, AT denote success rate, average number of steps, and average time (seconds), respectively.

Task	Sub Goal Num.	SR	AS	AT(s)	Eval Times
Craft a diamond shovel	15	18.75	23696.75	1184.84	64
Craft a diamond pickaxe	15	15.71	32189.50	1609.46	70
Craft a diamond axe	16	4.00	21920.50	1096.03	75
Craft a diamond hoe	15	4.61	24031.00	1201.55	65
Craft a diamond sword	15	14.52	27555.50	1377.78	62
Dig down and mine a diamond	15	9.09	20782.13	1039.11	64
Craft a jukebox	15	14.58	25056.00	1252.80	48

Table 11. The results of Optimus-1 on various tasks in the Redstone group. SR, AS, AT denote success rate, average number of steps, and average time (seconds), respectively.

Language Instruction	Sub-Goal Num.	SR	AS	AT(s)	Eval Times
Craft a piston	16	28.57	6457.10	822.85	35
Craft a redstone torch	16	29.63	6787.87	939.39	27
Craft an activator rail	18	15.68	8685.62	934.28	51
Craft a compass	23	15.00	14908.67	845.43	40
Craft a dropper	16	37.50	7272.80	1063.64	40
Craft a note block	16	24.32	6727.89	936.39	37

Table 12. The results of Optimus-1 on various tasks in the Armor group. SR, AS, AT denote success rate, average number of steps, and average time (seconds), respectively.

Task	Sub Goal Num.	SR	AS	AT(s)	Eval Times
Craft shield	14	43.33	7229.00	861.45	30
Craft iron chestplate	14	47.22	7230.24	851.51	36
Craft iron boots	14	23.81	6597.33	729.87	42
Craft iron leggings	14	6.67	9279.00	763.95	30
Craft iron helmet	14	58.14	6287.11	814.36	43
Craft diamond helmet	17	2.08	7342.00	867.10	48
Craft diamond chestplate	17	2.70	7552.00	777.60	37
Craft diamond leggings	17	9.68	7664.67	883.23	31
Craft diamond boots	17	16.67	10065.60	803.28	30
Craft golden helmet	17	12.50	11563.25	778.16	32
Craft golden leggings	17	14.60	10107.33	805.37	41
Craft golden boots	17	6.06	10311.00	915.55	33
Craft golden chestplate	17	9.67	10407.58	820.38	31